Software for Bibliographic Applications on Microcomputers: Where do we go from here?

by

Robert Valantin
Associate Director
Information Sciences Division
International Development Research Centre
Ottawa, Canada

Prepared for the

meeting on

Microcomputers and Bibliographic Information Systems in Latin America: Problems, Experiences and Projections

U.N. Economic Commission for Latin America
Santiago, Chile

24 - 27 April 1984

ABSTRACT

The predicament faced by information professionals and users wishing to acquire software for bibliographic applications on microcomputers is placed in perspective. The development of information management software is traced from mainframe to minicomputer environments, and factors to be considered in examining whether or not the situation is similar for microcomputers are identified. Some relevant characteristics of the microcomputer software industry and marketplace are presented. The need for the evaluation of commercially-available software is discussed, and inherent problems are identified. It is suggested that the current meeting should examine various approaches to some of these issues, and attempt to find the means for the on-going exchange of information on these topics.

Over the last decade, computers have radically changed the ways in which information is collected, processed, and disseminated. Terms like databases, data banks, on-line information retrieval, computerized bibliography production, telematics, and so on have become part of the working vocabulary of information professionals, rather than simply abstract concepts. Computer-based methods for handling information are receiving growing acceptance, and very few information systems being planned today are without a microchip buried somewhere in them.

Perhaps nothing is doing more to promote this trend than the advent of relatively powerful and inexpensive microcomputers to the scene. Just as in the days not very long ago when minicomputers began to make their mark in the field of information processing, by allowing centres to acquire stand-alone systems to permit them to process their information without being dependent upon large mainframe computer installations (whose priorities were often different), so too are microcomputers making their inroads today. Certainly it is not a question of one type of computer versus another, mainframe versus mini versus micro, since each has its place, its own economics, and its own advantages and disadvantages in various situations. Rather, it is the question of building "systems" from the various components available to meet the needs of information users.

But computers do require software in order to be useful. Software turns hardware into a working tool. In fact, for many information professionals and users the system is the software, or at least the software is the interface to nebulous processes being carried out somewhere down the line (be it in the little black box in front of them, the computer somewhere in the building, or perhaps even a continent away?).

Acquiring software has been a traditional problem for people wishing to use computers for information work. In the early days of experimentation with computers for information management, software was custom-written for the application at hand. Eventually, some of the programs whose design was reasonably general began to gain in popularity and were adopted (and often adapted) for use by others. In the commercial sector, this led to the development of "software packages" which could earn revenue in their own right, as well as making the hardware more accessible (and more purchasable) for users without the resources to carry out development themselves. To some degree, this also held true in the public sector, where software developed by an organization was

sought after by others with similar interests. A prime example of this is, of course, ISIS, which was developed by the ILO and is now maintained and distributed by Unesco. Centres, particularly in developing countries, which had access to mainframe IBM computers found it possible to use these computers for information work. The software was free of charge, training was available, the system was maintained on a continuing basis by an organization with an active interest in its use for development activities, and, best of all, the system was designed with information storage and retrieval applications in mind. Of course, there are other software packages as well (both commercial and non-commercial) available for use in information activities. Database software is big business now; $100 000 or more is not an unusual price to pay for mainframe database systems.

To some degree, history repeated itself with minicomputers. One of the first questions was: "Can minicomputers be used for information management in the same ways as mainframes could?". The first applications here were also custom-written. This was followed by the development of generalized software packages such as MINISIS, which was constructed by IDRC to fill the same user needs for minicomputers as ISIS does for mainframes. MINISIS fit into the ISIS family, and had the same advantages for developing countries as ISIS (no cost to public-sector institutions in developing countries, training, support, and a design with the bibliographic information application in mind). Of course, now there are quite a number of other packages for use on minicomputers as well.

Software for mainframes and minicomputers tends to be written for a particular piece of hardware or series of machines produced by a single manufacturer. In some cases, of course, there are versions of software which run on different machines, but this appears to be the exception rather than the rule. Because most manufacturers have written or adopted their own operating systems, which are usually incompatible, the question of transportability of software has not been well resolved. It is not surprising that software producers have tried to recoup their costs and make a "profit" within families of hardware, since there are relatively high costs involved (either explicitly in terms of commercial costs or implicitly in terms of development and maintenance costs borne by non-profit organizations).

Now microcomputers have arrived. Because of their low cost, ready availability, immediate usefulness as sophisticated word processors

or calculators, the general trend towards "user-friendliness", and strong advertising and promotion, information managers have been eager to look to microcomputers to provide solutions to many of their current information problems and future information needs.

The question is: "Will the pattern established by the use of mainframes and minicomputers for handling information repeat itself here?".

A number of factors lead to a serious examination of this question. First of all, with mainframes and minicomputers the twin concepts of "database management systems" and "software packages" have taken hold. Thus, people have begun to think in terms of using packages wherever reasonable, or at least adapting them as necessary, rather than necessarily sitting down to write applications from scratch. In many cases, packages simply provide the building blocks out of which systems can be constructed.

Programming languages and techniques have become much more sophisticated, and the marketplace has responded with fourth generation languages and system development tools. In some cases, packages provide command languages which are programming languages in their own right, and it sometimes is difficult to differentiate between application building and programming. A good deal of standardization with respect to operating systems software has taken place with microcomputers, thus encouraging wider marketability of software products.

One might expect software for microcomputers to cost less to produce than software for mainframes and minicomputers. Although hardware has become much cheaper (and is a capital cost in software development), and although sophisticated tools are available to facilitate the development of systems, labour has not decreased in cost. Furthermore, users are conditioned to expect better and more sophisticated systems. Thus the economics dictate that software developers have to look to sell many copies of their systems across a wide variety of hardware in order to make a profit. Still, the lure of a rather large and expanding marketplace is very great, and a whole host of software developers have begun producing software. The software industry has come into its own here too. In addition, some manufacturers have started to convert software which ran on mainframes and minicomputers to operate on microcomputers. As hardware develops in power and sophistication, it becomes possible to consider running software on "micro mainframes". Technical advances make

it possible to consider putting a 100 Mbyte disk on a $10 000 microcomputer (at least in a shared storage or network environment).

Thus, users wishing to use microcomputers find themselves in a somewhat different environment, faced with a wide variety of software which is said to be applicable for their needs. Certainly, the gradual acceptance of information as a commodity and the "information revolution" has helped sensitize the software production sector. But it is precisely because of this bewildering variety of software and the often conflicting claims that users are questioning how to proceed. Users have become very sophisticated and demanding, and somehow the packages being offered never quite seem to meet their needs -- or do they?

Evaluating the currently available software is one logical first step in a response to this predicament. Within the development community, certainly there are great expectations for the usefulness of microcomputers to provide many information management functions. In particular, the growing number of information networks and "cooperative information systems" require adequate tools for the processing and dissemination of information, both at the local and more central levels. Some users have experimented with a number of basic database packages and have succeeded in using them to meet limited information processing requirements. Others have even experimented with writing simple programs, often in BASIC. But how can users decide if this pool of commercially-available software is useful for their needs?

Of course, the question is not a simple one. Different users have different requirements. Different users expect or need (and there is a difference between the two) different levels of sophistication in their systems. Even when things are narrowed down to "bibliographic systems", there is a wide discrepancy in the complexity of operations required. It seems clear that any evaluation methodology must take into account the specific application for which the software will be used. It seems unlikely that it will be possible to rate software for its usefulness across a wide variety of applications, since a package which is useful in one environment may prove inadequate in another.

Various approaches to these issues will be discussed at this meeting. There will be presentations on guidelines on the evaluation of commercial software for bibliographic applications, on one institution's experience in carrying out such evaluations,

and on several users' experiences with commercial packages. There will also be presentations on the development of portable software specifically designed to meet the bibliographic information management needs of users, on the design of software, and on several approaches to and experiments with implementing systems. In addition, the questions of hardware requirements and data exchange will be addressed.

It is not likely that a single approach to this problem will meet everyone's needs now and in the future. What is important, and what this meeting should achieve, is the exchange of information on these efforts so that the results of this research and experimentation are available to those who can most benefit from them. With a little cooperation and planning, it should be possible to put microcomputers to effective use in information applications in Latin America - which is, after all, why we are all here this week.